



25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM  
2014

## Multimedia Teaching Aid for Students of Basics of Control Theory in MATLAB and SIMULINK

Perutka Karel\*, Zaoral Tomas

*Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam. T.G.M. 5555, 760 01 Zlin, Czech Republic*

---

### Abstract

The paper deals with the multimedia teaching aid which was created for students of the Bachelor curriculum entitled as Information and control technology. It helps mainly students of distance curriculum to study the slight introduction to control theory and programming in MATLAB from their home. They could run the DVD at home and watch how the controller is programmed or created.

© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of DAAAM International Vienna

*Keywords:* Control in education; control theory; MATLAB; PID control

---

### 1. Introduction

#### 1.1. General Introduction

The application of new educational methods, aids and tools in the area of control is important aspect of modern educational system during last years also we are in the era of decrease of public funding [4], especially the global view of graduate process should be taken into account [6]. Nice survey about control was written by Åström and Kumar [7]. There is important to include technical and non-technical issues in control education [3]. There are several applications and tools focused on education control, for example experimental educational program for computer control of control system [5], or it is realized even in real time [8] where distance learning laboratories play important role [9] or it can be realized as application of virtual tool for control education [10].

---

\* Corresponding author. Tel.: +420 576035202; fax: +420 576035279.  
*E-mail address:* [kperutka@fai.utb.cz](mailto:kperutka@fai.utb.cz)

From the area of model predictive control was created nice package for undergraduate education [2]. Nice application in the area of education was realized in Spain [11], [12].

### *1.2. Matlab and Simulink Environment*

All the following text is based on the work of Zaoral which created under the supervision of Perutka [10]. MATLAB is an integrated environment for scientific computing, modeling, algorithms design, simulation, analysis and presentation of data parallel computations, measurement and signal processing, design of control and communication systems. MATLAB computing system in recent years has become the global standard for engineering calculations and simulations in science, research, industry and education. MATLAB provides its users with not only a powerful graphical and computational tools but also an extensive specialized library functions with powerful fourth-generation programming language. Libraries are thanks to their range usable in virtually all areas of human activity. With its architecture MATLAB is designed for those who need to solve numerically intensive tasks while they do not want or do not have time to examine the nature of mathematical problems. More than one million users worldwide use the MATLAB software, which is much easier than, say, Fortran or C and which has great potential productivity and creativity. MATLAB strongest site is considered extremely fast computing core with optimal algorithms that are proven through years of service on top worldwide locations. MATLAB has been implemented on all major platforms (Windows, Linux, Solaris, Mac). Name MATLAB is an abbreviation of the words MATrix Laboratory (loosely translated as "laboratory with matrices"), which reflects the fact that key data structure in the calculations in MATLAB are matrices. Custom programming language is based on the language Fortran.

### *1.3. Introduction about Simulink*

Simulink is a software package that allows you to model, simulate and analyze systems whose outputs change over time. Such systems are often referred to as dynamic systems. For modeling, Simulink provides a graphical user interface (GUI), in which the user draws up its own model of each block library using the click-and-drag mouse operations, a block clicking and dragging on the block where the block in the model is located. Simulink includes a comprehensive block library, such as library resources, library of linear and nonlinear elements and many others. Furthermore, the user can also create his/her own blocks. Simulink can be used to study the behavior of a wide range of dynamic real-world systems, including electrical, mechanical and thermodynamic systems. The first Simulink edition contained MATLAB version 3.5 in 1990.

## **2. Control theory used for video-guide creation**

### *2.1. Ziegler-Nichols method*

This is a heuristic method of tuning PID controller, which is credited by John G. Ziegler and Nathaniel B. Nichols. According to these authors, this method was named after their surnames. The method has three modifications, of which the second modification is used in video tutorials.

#### **Ziegler-Nichols method of critical parameters (second modification)**

This method is applicable for linear continuous and discrete control systems. The principle consists of bringing the circuit to the critical state, i.e., the oscillating stability limit, wherein the controller operates with only proportional element. Integral and derivative component is excluded. Reaches a circuit stability limit, the gain value at which this condition occurred, is called the critical gain  $K_{pk}$ . The period of oscillation at which this occurred, then we call the critical period of oscillation  $T_k$ .

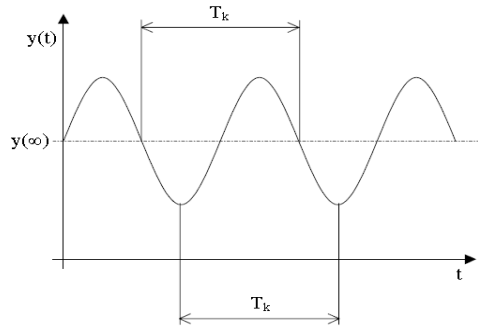


Fig. 1. Determination of the critical period.

The following table will allow us to determine the values of the controller parameters:

Table 1. Ziegler-Nichols (second modification) relations for calculating the parameters of the controller.

Controller	$T_P$	$T_I$	$T_D$
<b>P</b>	$0.5 K_{pk}$		
<b>PI</b>	$0.45 K_{pk}$	$0.83 T_k$	
<b>PD</b>	$0.4 K_{pk}$		$0.5 T_k$
<b>PID</b>	$0.6 K_{pk}$	$0.5 T_k$	$0.125 T_k$

## 2.2. Åström controller

Åström algorithm for calculating the output level of the discrete PID controller is given by

$$u(k) = u_{PI}(k) + u_D(k),$$

where

$$u_{PI}(k) = K_p \cdot [y(k-1) - y(k)] + \frac{K_p T_0}{2T_I} \cdot [e(k) + e(k-1)] + \beta K_p \cdot [w(k) - w(k-1)] + u_{PI}(k-1),$$

and

$$u_D(k) = K_p \cdot \frac{T_D \alpha}{T_D + T_0 \alpha} \cdot [y(k-1) - y(k)] + \frac{T_D}{T_D + T_0 \alpha} \cdot u_D(k-1).$$

Parameters in previous equations are chosen from the intervals  $\beta \in (0;1)$ ,  $\alpha \in \langle 3;20 \rangle$ .

## 2.3. Runge-Kutta method

The most widely used method for the numerical solution of the ordinary differential equations. It is characterized by fourth order accuracy. At each step are calculated four auxiliary variables. This technique was developed about 1900 by German mathematic C. Runge and M. W. Kutta. Improved Runge-Kutta methods were suggested in 1970 by the German mathematician Erwin Fehlberg and this enhancement is given the name of the method as Fehlberg or also Runge-Kutta-Fehlberg method.

### Runge-Kutta-Fehlberg method.

Enhancement is in the sense that each step can be controlled by the step size and thus the resultant errors using two methods. Fehlberg found these two variants of Runge-Kutta methods, which require the simultaneous use of only 6 operations. This method is of the fifth order accuracy.

Fehlberg used both Runge-Kutta method of fifth order in which the function value  $y_{n+1}$  is expressed as

$$y_{n+1} = y_n + \alpha_1 k_1 + \alpha_2 k_2 + \alpha_3 k_3 + \alpha_4 k_4 + \alpha_5 k_5 + \alpha_6 k_6,$$

which could be written as

$$y_{n+1} = y_n + \alpha^T k,$$

where

$$\alpha^T = \left[ \frac{16}{135}, 0, \frac{6656}{12825}, \frac{28561}{56430}, -\frac{9}{50}, \frac{2}{55} \right],$$

and fourth order method

$$y_{n+1}^* = y_n + \beta_1 k_1 + \beta_2 k_2 + \beta_3 k_3 + \beta_4 k_4 + \beta_5 k_5,$$

so

$$y_{n+1}^* = y_n + \beta^T k,$$

where

$$\beta^T = \left[ \frac{25}{216}, 0, \frac{1408}{2565}, \frac{2197}{4104}, -\frac{1}{5} \right].$$

The components of the vector are given by the following equations:

$$\begin{aligned} k_1 &= h \cdot f(x_n, y_n), \\ k_2 &= h \cdot f\left(x_n + \frac{h}{4}, y_n + \frac{k_1}{4}\right), \\ k_3 &= h \cdot f\left(x_n + \frac{3h}{8}, y_n + \frac{3k_1}{32} + \frac{9k_2}{32}\right), \\ k_4 &= h \cdot f\left(x_n + \frac{12h}{13}, y_n + \frac{1932k_1}{2197} - \frac{7200k_2}{2197} + \frac{7296k_3}{2197}\right), \\ k_5 &= h \cdot f\left(x_n + h, y_n + \frac{439k_1}{216} - 8k_2 + \frac{3680k_3}{512} - \frac{845k_4}{4104}\right), \\ k_6 &= h \cdot f\left(x_n + \frac{h}{2}, y_n - \frac{8k_1}{27} + 2k_2 - \frac{3544k_3}{2565} + \frac{1859k_4}{4104} - \frac{11k_5}{40}\right). \end{aligned}$$

Estimation error is counted according to

$$\varepsilon_{n+1} \approx y_{n+1} - y_{n+1}^* = \frac{k_1}{360} - \frac{128k_3}{4275} - \frac{2197k_4}{75240} + \frac{k_5}{50} + \frac{2k_6}{55}.$$

### 3. Created video-guide for students of control theory

To create a video guide the Camtasia Studio 7 program was used. It is a program designed to create video guide that allows you to record screen and edit videos and create DVD menu.

### 3.1. Camtasia Studio 7 environment

After running, the desktop of the program is displayed. The work area is divided into three sub-sections. On the left, there is an interactive window, which displays the currently selected tool for video editing. On the right, there is a screen that is used to edit video and also to view it. At the bottom of the screen, there is the timeline.

The main tools included in Camtasia Studio 7 are:

- descriptions,
- zooming,
- transitions,
- voice recording
- cursor effects,
- adjustment of audio tracks,
- subtitles

### 3.2. Creating DVD menu

To create a DVD menu, the Camtasia MenuMaker was used (Figure 2). This application is a part of the Camtasia Studio 7 and it runs through the menu option Tools -> Studio Tools -> Camtasia MenuMaker. After starting the application, the Welcome dialog appears. It offers options such as creating a new project using the wizard, open a new project, open an existing project or open the last project. After selecting the first or second option, the basic library of 11 templates opens. The user can select one of the available templates or apply the template and select "Blank template". This opens a new project that can be edited using the Properties Menu. Window Menu Properties contains four tabs: General, Cursor, List and Content. In the General tab you can set the caption menu, background color or background image and canvas size menu. In the Cursor tab, you can set the appearance of the cursor, the sound of the cursor when you click a sound and when you move the cursor over the object. In the List tab, you can set the text color (normal, visited, highlighted), text alignment, frame style menu and position the frame. The last tab Content is the most important. You create links to the video files that you want to appear in the menu when you click on the link.

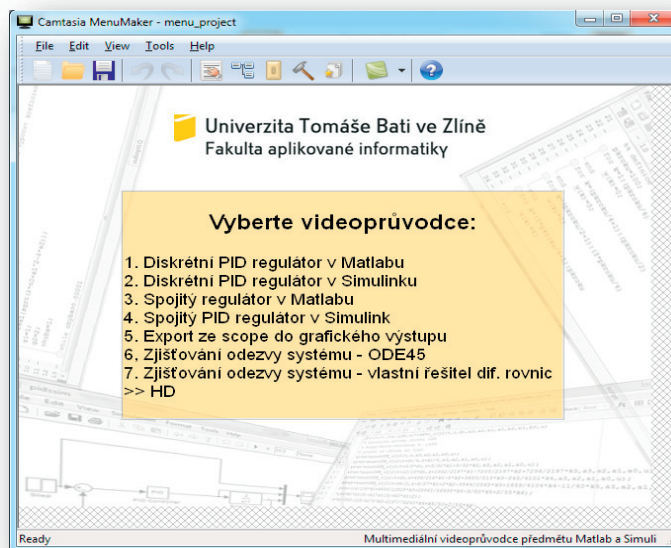


Fig. 2. Camtasia Menu Maker.

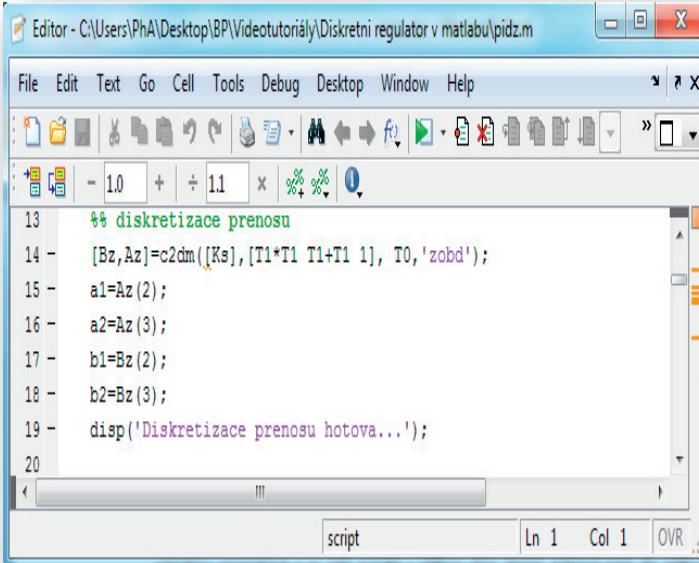
### 3.3. Running the DVD

Menu, which is located on the DVD, runs automatically when the DVD is inserted into your computer. If it does not, this menu can be run manually by opening the contents of a DVD file and by running the file named as “videoprůvodce.exe”. Menu is realised easily and intuitively (Figure 2). It contains title and seven links. Link on which is the mouse cursor is highlighted by turquoise color. After clicking the mouse on the selected link, the selected video starts from the video guide in video format 4:3 with a resolution of 720x480 pixels. The resolution is set to optimize for a wide range of monitors or projectors. After clicking the mouse on the text HD, which is located in the last section menu, a sub-menu is opened. This sub-menu is used to run individual videos from video guide in widescreen 16:9 widescreen format, at a resolution of 1280x720. The widescreen video format video quality is higher than the classical 4:3 format.

## 4. Short description of created video guides

### 4.1. Discrete PID controller in MATLAB

The user is acquainted with discrete PID controller using MATLAB script, which is divided into six sub-sections. At the beginning, the screen of the command window is cleared, from the Workspace are deleted all the variables and the disp command is used here and that displays text in the command window. Furthermore, the transfer function of the system is defined together with a sampling period. This followed by the discretization of the transfer function (Figure 3) using C2DM function and the backward rectangular method (ZOBD) is used here.



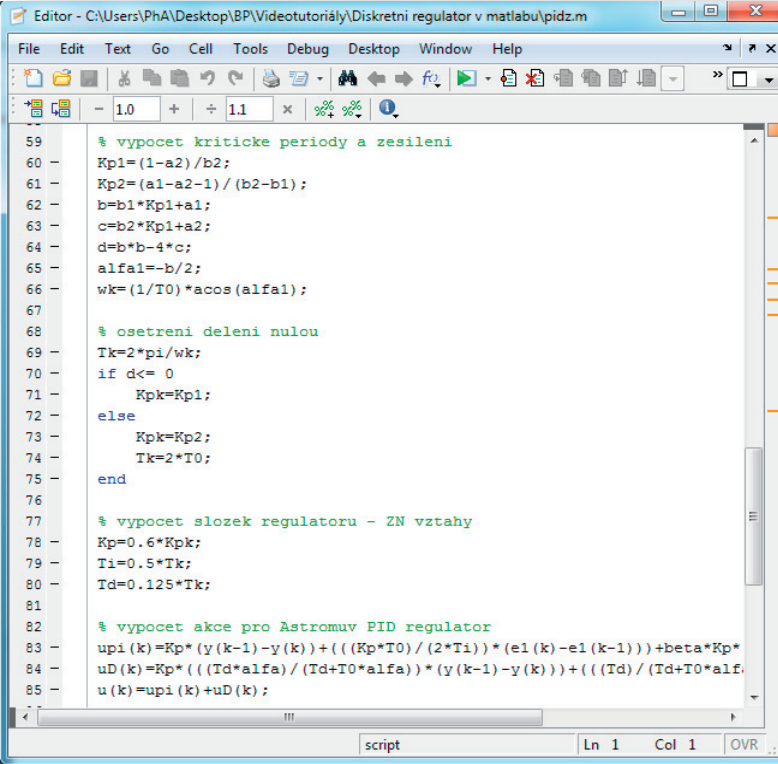
```

Editor - C:\Users\PhA\Desktop\BP\Videotutoriály\Diskretni regulator v matlabu\pidz.m
File Edit Text Go Cell Tools Debug Desktop Window Help
13 %% diskretizace prenosu
14 - [Bz,Az]=c2dm([Ks],[T1*T1 T1+T1 1], T0,'zobd');
15 - a1=Az(2);
16 - a2=Az(3);
17 - b1=Bz(2);
18 - b2=Bz(3);
19 - disp('Diskretizace prenosu hotova...');
20
script Ln 1 Col 1 OVR

```

Fig. 3. Sample source code - discretization of transfer function.

Then, the value of the desired value is defined using for four cycles and the variable with which it continues to operate are initialized. The following is the most important part of the script, it is used computing cycle of the controller, which is also in the for loop. This includes the calculation of critical periods and critical gain, treatment of division by zero, the calculation of the parameters of the controller with the Ziegler-Nichols relations and counting the action signal value for calculation Aström PID controller. The last part of the script takes care of drawing the results of process control.



```

59  % vypočet kritické periody a zesílení
60  Kp1=(1-a2)/b2;
61  Kp2=(a1-a2-1)/(b2-b1);
62  b=b1*Kp1+a1;
63  c=b2*Kp1+a2;
64  d=b*b-4*c;
65  alfa1=-b/2;
66  wk=(1/T0)*acos(alfa1);
67
68  % ošetření dělení nulou
69  Tk=2*pi/wk;
70  if d<= 0
71      Kpk=Kp1;
72  else
73      Kpk=Kp2;
74      Tk=2*T0;
75  end
76
77  % vypočet složek regulátoru - ZN vztahy
78  Kp=0.6*Kpk;
79  Ti=0.5*Tk;
80  Td=0.125*Tk;
81
82  % vypočet akce pro Astromův PID regulátor
83  upi(k)=Kp*(y(k-1)-y(k))+((Kp*T0)/(2*Ti))*(e1(k)-e1(k-1))+beta*Kp*
84  uD(k)=Kp*((Td*alfa)/(Td+T0*alfa))*(y(k-1)-y(k))+((Td)/(Td+T0*alf
85  u(k)=upi(k)+uD(k);

```

Fig. 4. Sample source code – main cycle.

#### 4.2. Discrete PID controller in Simulink

In this video, the user will learn how to create a model of a discrete PID controller in Simulink. Since this is the first video in which the student operates in Simulink and are therefore at the beginning there are shown two ways how to run Simulink. After running Simulink, a new window is created and we can start to create a model of a discrete PID controller. Firstly a diagram of the controller is created and the scheme is applied to mask the subsystem. After you create the subsystem, you insert additional blocks from Simulink libraries to build the model of discrete PID controller. In the last part of the video the Step block is changed for block Repeating Sequence, which defines the actual course of the requested value.

This video guide uses the following blocks:

- step,
- repeating sequence,
- sum,
- unit delay,
- gain,
- zero-order hold,
- transfer fcn,
- mux,
- scope.

#### 4.3. Continuous-time controller in MATLAB

This video guide consists of five M - files:

- rkf4r\_c2j3contr.m ,
- soustODR\_c2j3contr.m ,
- soustODR\_c0j2.m ,
- suboptLQ.m and
- AregulaceLQ.m .

The first four M-files are functions which are used in the main script AregulaceLQ.m. At the beginning of the main script (AregulaceLQ.m) the variables from the Workspace are erased and the global variables and control time are defined. Simulation time is set at 1500, but in reality, the total control time is 10 times lower, i.e. 150 seconds. It is because of the implementation of continuous-time domain. In another part of the script, the start time and end time vectors and time step are defined. Then variables are initialized, which are used in other parts of the script to work. To calculate the parameters of the controller is first necessary to define the parameters for the initialization of the controller parameters to initialize the system and weighting coefficients. Then the function suboptLQ is used to calculate the parameters of the controller. Furthermore, the controller output is counted, for which the rkf4r\_c2j3contr function is used and it serves as a separate solver of differential equation. Finally, the control error is counted. At the end of the main script, the function to draw the history of control is used.

#### 4.4. Continuous-time PID controller in Simulink

Users will learn how to create a model of continuous-time PID controller in Simulink. Model is first assembled from individual blocks of Simulink library, then the mask of the subsystem is applied and in the last part of the video guide, the subsystem, which consists of the controller, is replaced by the special block called PID Controller, which replaces the PID controller.

### Conclusion

The paper presents created multimedia teaching aid for students of Bachelor curricula. It is a set of MP4 files describing the creation of sample controllers in MATLAB and SIMULINK with Czech audio and source codes located in the DVD with its own menu. This teaching aid is positively evaluated by the students as a nice introduction in programming the application from the area of control theory. Future work will be oriented to enlarge the set of video guides and to prepare other version of audio files in other languages.

### References

- [1] Zaoral, T., 2011. Multimedia video-guide of basics of control theory in Matlab and Simulink for “Matlab and Simulink” subject. Thesis, Faculty of Applied Informatics, Tomas Bata University in Zlin, Czech Republic (in Czech).
- [2] Richmond, P., Chen, D., 2012. A model predictive control package for undergraduate education. *Education for Chemical Engineers*, 7, e43-e50.
- [3] Brandt, D., Imamichi, C., McGregor, H., Moses, I., van der Vorst, R., 1996. Integrating technical and non-technical issues in control education. *Control Eng. Practice*, 4, 655-662.
- [4] Viaene, J.-M., Zilcha, I., 2013. Public funding of higher education. *Journal of Public Economics*, 108, 78-89.
- [5] Jung, J.H., Lee, M., Lee, J., Han, Ch., 2000. A development of experimental education program: computer control of multi-stage level control system. *Computers and Chemical Engineering*, 24, 1497-1502.
- [6] Edgar, T.F., Ogunnaike, B.A., Muske, K.R., 2006. A global view of graduate process control education. *Computers and Chemical Engineering*, 30, 1763-1774.
- [7] Åström, K.J., Kumar, P.R., 2014. Control: A perspective. *Automatica*, 50, 3-43.
- [8] Mahoney, D., Young, B., Svrcek, W., 2000. A completely real time approach to process control education for process systems engineering students and practitioners. *Computers and Chemical Engineering*, 24, 1481-1484.
- [9] Stefanovic, M., 2013. The objectives, architectures and effects of distance learning laboratories for industrial engineering education. *Computers and Education*, 69, 250-262.



- [10] Savaş, K., Erdal, H., 2010. Automatic control simulation environment system (ACSES) designed as a virtual tool for control education. *Procedia Social and Behavioral Sciences*, 2, 5233-5237.
- [11] Matin-Villalba, C., Urquia, A., Dormido, S., 2012. Development of virtual-labs for education in chemical process control using Modelica. *Computers and Chemical Engineering*, 39, 170-178.
- [12] Mendéz, J.A., González, E.J., 2013. A control system proposal for engineering education. *Computers and Education*, 68, 266-274.